Discriminative Metric Learning in Nearest Neighbor Models for Image Annotation

Matthieu Guillaumin, Thomas Mensink,
Jakob Verbeek & Cordelia Schmid

LEAR Team, INRIA Rhône-Alpes
Grenoble, France

Discriminative Metric Learning in Nearest Neighbor Models for Image Annotation

- **Goal:** predict relevant keywords for images
- **Approach:** generalize from a data base of annotated images
- **Application 1: Image annotation**
  - Propose a list of relevant keywords to assist human annotator
- **Application 2: Keyword based image search**
  - Given one or more keywords propose a list of relevant images

Examples of Image Annotation

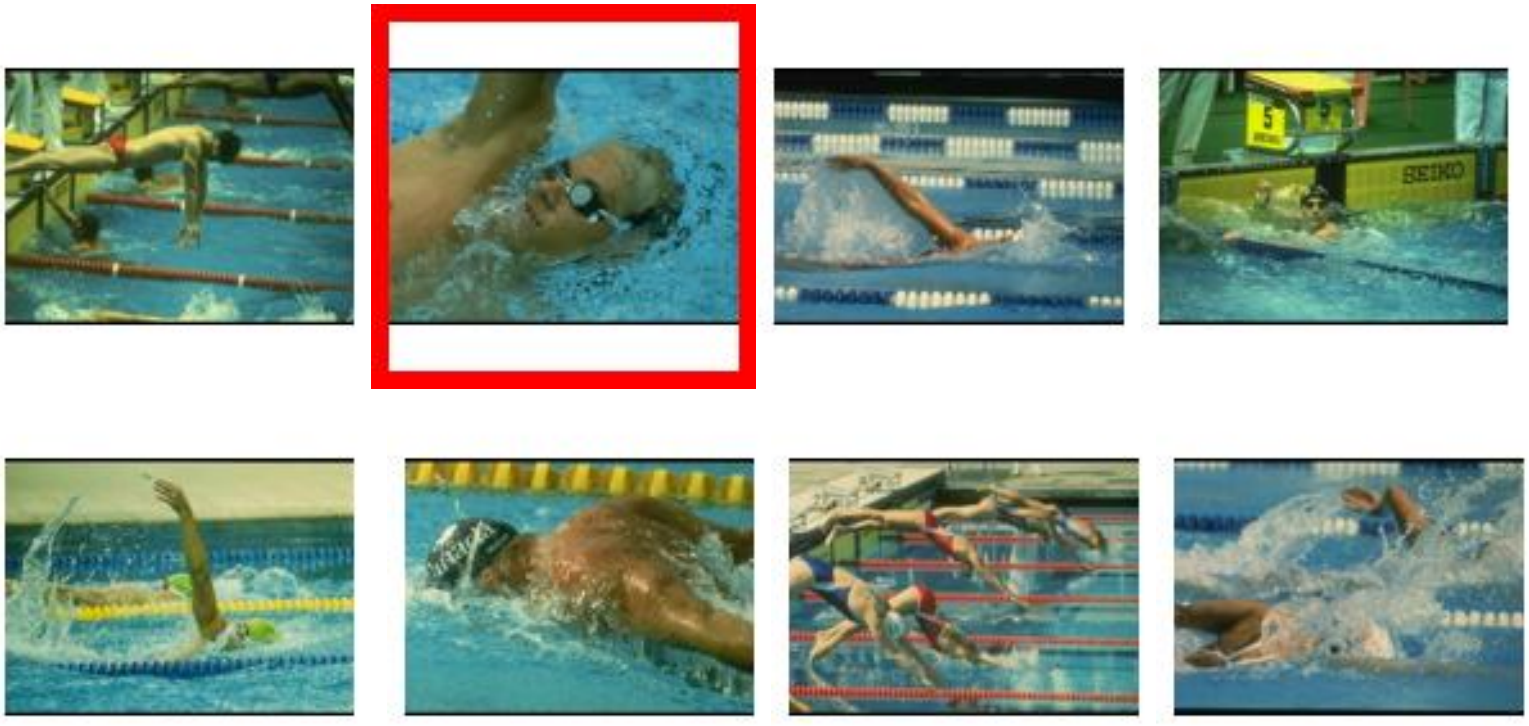
true		**predicted** (confidence)
glacier		glacier (1.00)
mountain		mountain (1.00)
people		front (0.64)
tourist		sky (0.58)
		people (0.58)

Examples of Image Annotation

true		**predicted** (confidence)
landscape		llama (1.00)
lot		water (1.00)
meadow		landscape (1.00)
water		front (0.60)
		people (0.51)

Examples of Keyword Based Retrieval

- **Query:** water, pool
- **Relevant images:** 10
- **Correct:** 9 among top 10



Examples of Keyword Based Retrieval

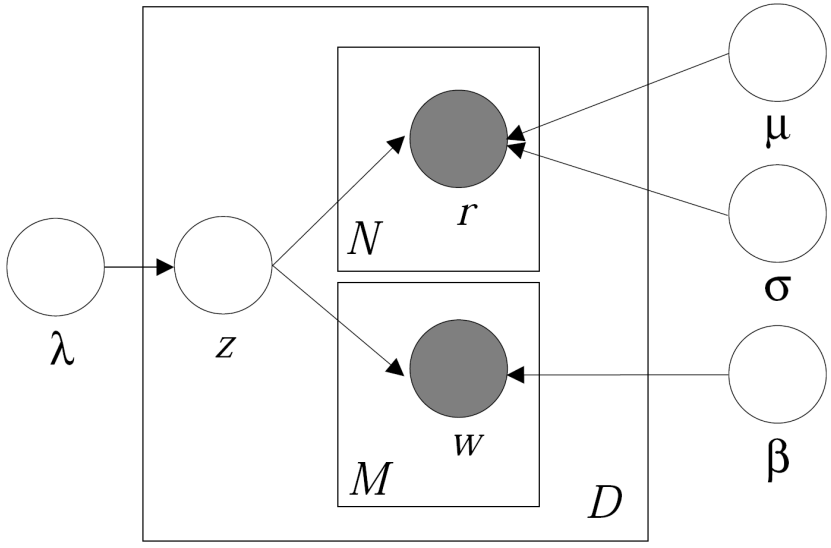
- **Query:** beach, sand
- **Relevant images:** 8
- **Correct:** 2 among top 8

Presentation Outline

1. **Related work**
2. **Metric learning for nearest neighbors**
3. **Data sets & Feature extraction**
4. **Results**
5. **Conclusion & outlook**

Related Work: Latent Topic Models

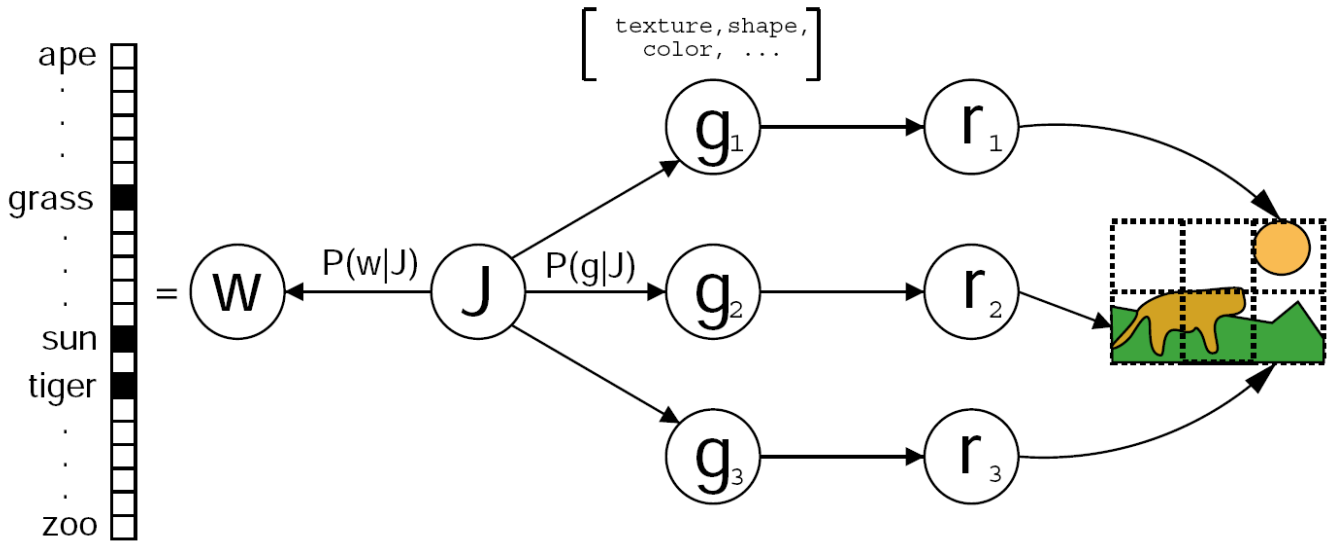
- **Inspired from text-analysis models**
  - Probabilistic Latent Semantic Analysis
  - Latent Dirichlet Allocation
- **Generative model over keywords and image regions**
  - Trained to model both text and image
  - Condition on image to predict text
- **Trade-off: overfitting & capacity limited by nr. of topics**



[Barnard et al., ”Matching words and pictures”, JMLR’03]

Related Work: Mixture models approaches

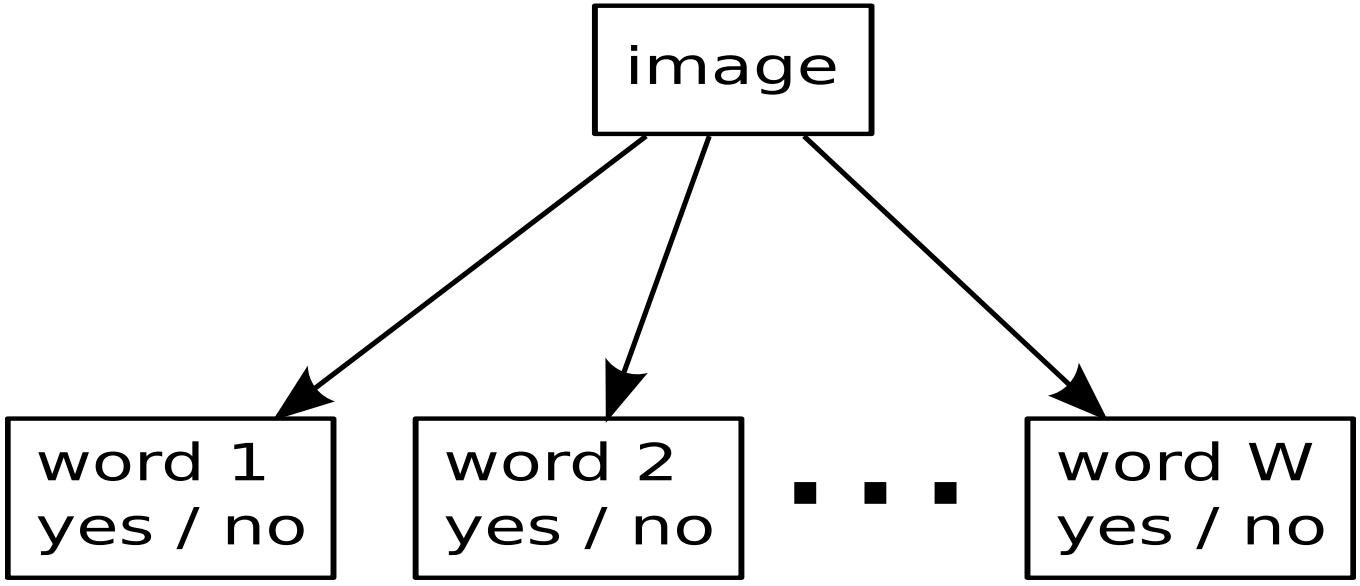
- **Generative model over keywords and image**
  - Kernel density estimate (KDE) over image features
  - KDE gives posterior weights for training images
  - Use weights to average training annotations
- **Non-parametric model**
  - only need to set KDE bandwith



[Feng et al., "Multiple Bernoulli relevance models", CVPR'04]

Related Work: Parallel Binary Classifiers

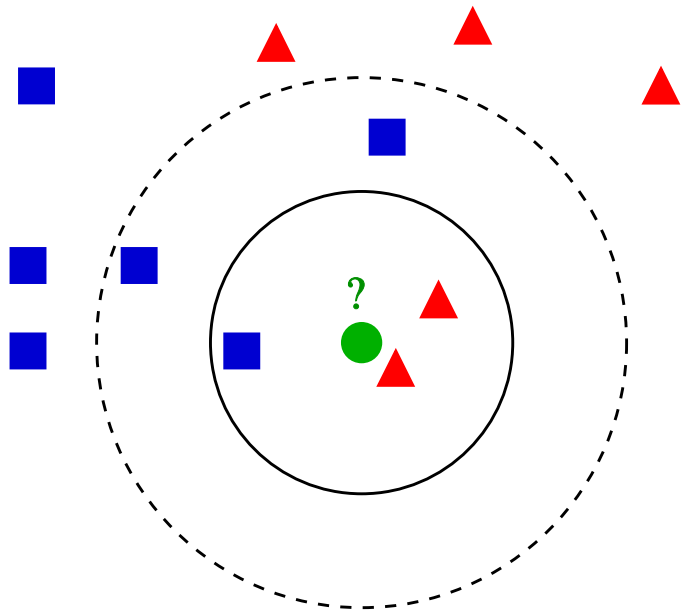
- **Learn a binary classifier per keyword**
  - Need to learn many classifiers
  - No parameter sharing between keywords
- **Large class imbalances**
  - 1% positive data per class no exception



[Grangier & Bengio. ”A discriminative kernel-based model to rank images from text queries”, PAMI’08]

Related Work: Local learning approaches

- **Use most similar images to predict keywords**
  - Diffusion of labels over similarity graph
  - Nearest-neighbor classification
- **State-of-the-art image annotation results**
- **What distance to define neighbors?**



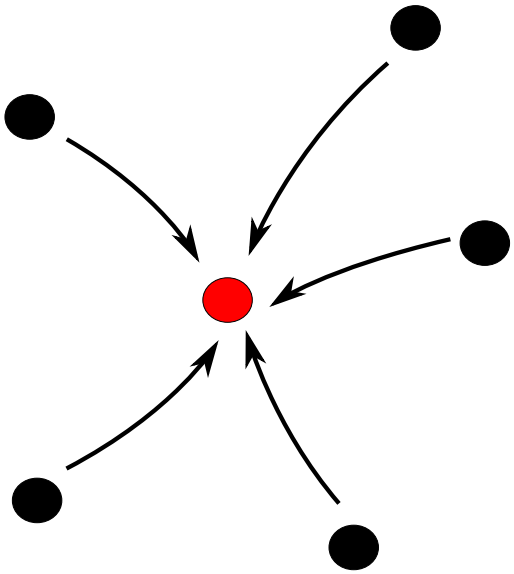
[Makadia et al., ”A new baseline for image annotation”, ECCV’08]

Presentation Outline

1. Related work
2. **Metric learning for nearest neighbors**
3. Data sets & Feature extraction
4. Results
5. Conclusion & outlook

A predictive model for keyword absence/presence

- **Given: relevance of keywords w for images i**
  - $y_{iw} \in \{-1, +1\}, \quad i \in \{1, \ldots, I\}, \quad w \in \{1, \ldots, W\}$
- **Given: visual dissimilarity between images**
  - $d_{ij} \geq 0, \quad i, j \in \{1, \ldots, I\}$
- **Objective:** optimally predict annotations y_{iw}

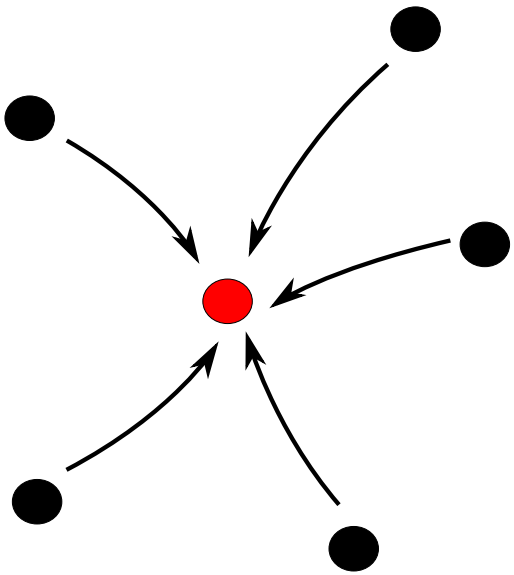


A predictive model for keyword absence/presence

- π_{ij} : **weight of train image j for predictions for image i**
  - Weights defined through dissimilarities
  - $\pi_{ij} \geq 0$ and $\sum_j \pi_{ij} = 1$

$$p(y_{iw} = +1) = \sum_j \pi_{ij}\, p(y_{iw} = +1|j) \tag{1}$$

$$p(y_{iw} = +1|j) = \begin{cases} 1 - \epsilon & \text{for } y_{jw} = +1 \\ \epsilon & \text{otherwise} \end{cases} \tag{2}$$



A predictive model for keyword absence/presence

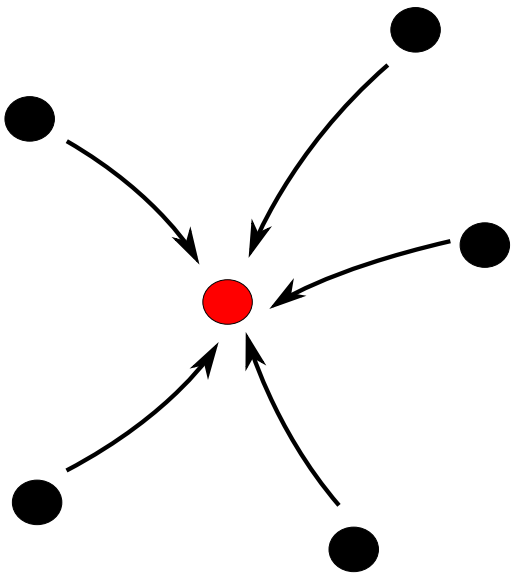
- **Parameters:** definition of the π_{ij} from visual similarities

$$p(y_{iw} = +1) = \sum_j \pi_{ij}\, p(y_{iw} = +1|j)$$

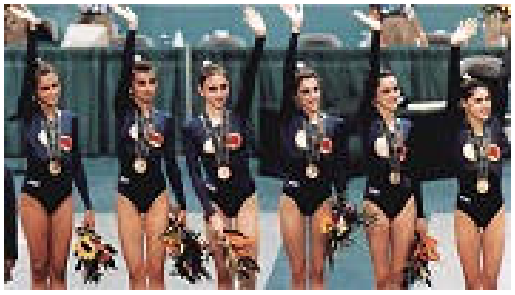
- **Learning objective:** maximize probability of actual annotations

$$\mathcal{L} = \sum_i \sum_w c_{iw} \ln p(y_{iw}) \tag{3}$$

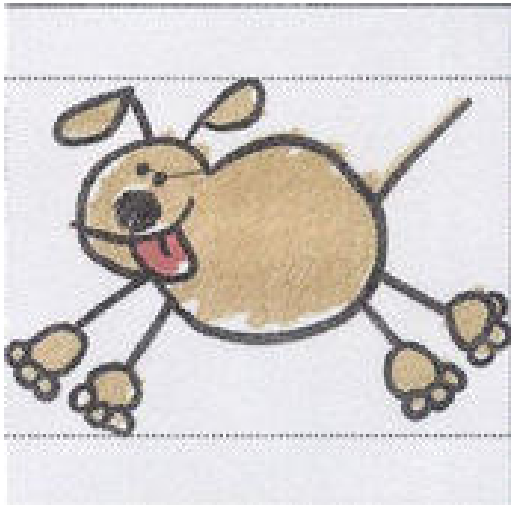
- **Annotation costs:** absences are much 'noisier'
  - Emphasise prediction of keyword presences



Example-absences: examples of typical annotations



Actual: **wave** *(0.99)*, **girl** *(0.99)*, flower *(0.97)*, black *(0.93)*, america *(0.11)*
Predicted: people *(1.00)*, woman *(1.00)*, **wave** *(0.99)*, group *(0.99)*, **girl** *(0.99)*



Actual: **drawing** *(1.00)*, **cartoon** *(1.00)*, kid *(0.75)*, dog *(0.72)*, brown *(0.54)*
Predicted: **drawing** *(1.00)*, **cartoon** *(1.00)*, child *(0.96)*, red *(0.94)*, white *(0.89)*

Rank-based weighting of neighbors

- **Weight given by rank**
  - The k-th neighbor always receives same weight
  - If j is k-th nearest neighbor of i

$$\pi_{ij} = \gamma_k \tag{4}$$

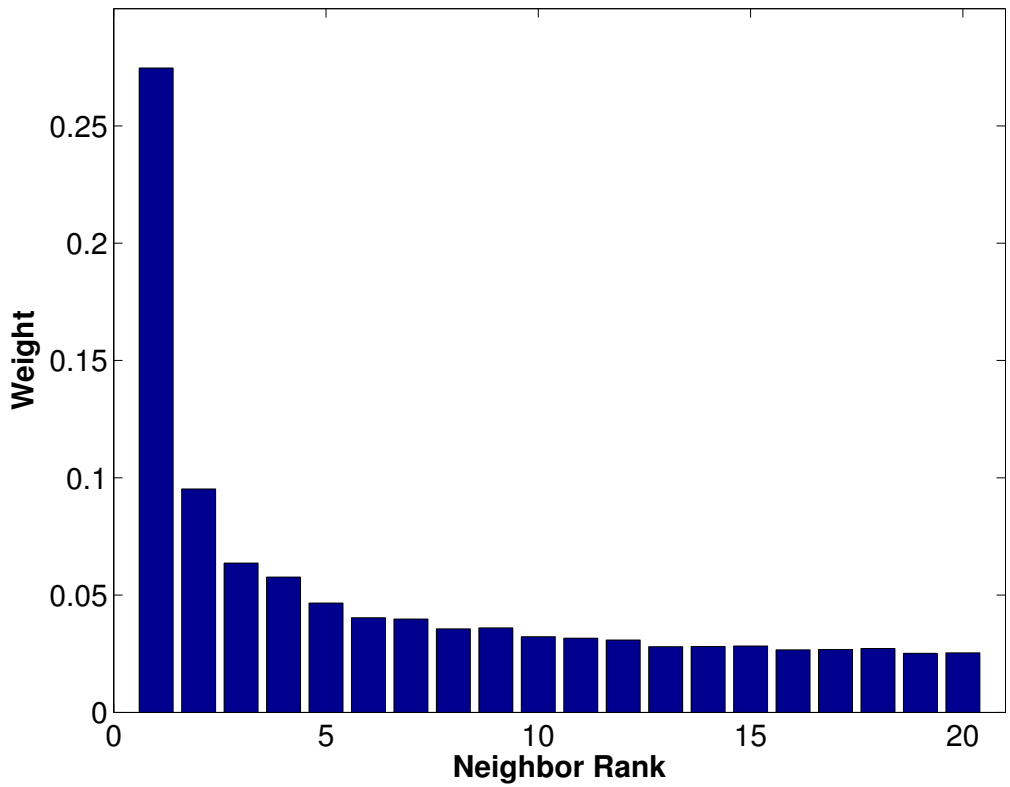
- **Optimization:** $\mathcal{L}$ concave with respect to $\{\gamma_k\}$
  - Expectation-Maximization algorithm
  - Projected gradient descent

$$p(y_{iw} = 1) = \sum_j \pi_{ij}\, p(y_{iw} = 1|j)$$

$$\mathcal{L} = \sum_i \sum_w c_{iw} \ln p(y_{iw})$$

Rank-based weighting of neighbors

- **Effective neighborhood size set automatically**



Distance-based weighting of neighbors

- **Weight given by distance:** d_{ij} visual distance between images

$$\pi_{ij} = \frac{\exp(-\lambda d_{ij})}{\sum_k \exp(-\lambda d_{ik})} \tag{5}$$

- **Single parameter:** controls weight decay with distance
  - Weights are smooth function of distances

- **Optimization:** gradient descent

$$\frac{\partial \mathcal{L}}{\partial \lambda} = W \sum_{i,j} (\pi_{ij} - \rho_{ij}) d_{ij} \tag{6}$$

$$\rho_{ij} = \frac{1}{W} \sum_w p(j|y_{iw}) = \frac{1}{W} \sum_w \frac{\pi_{ij} p(y_{iw}|j)}{\sum_k \pi_{ik} p(y_{iw}|k)} \tag{7}$$

Metric learning for nearest neighbors

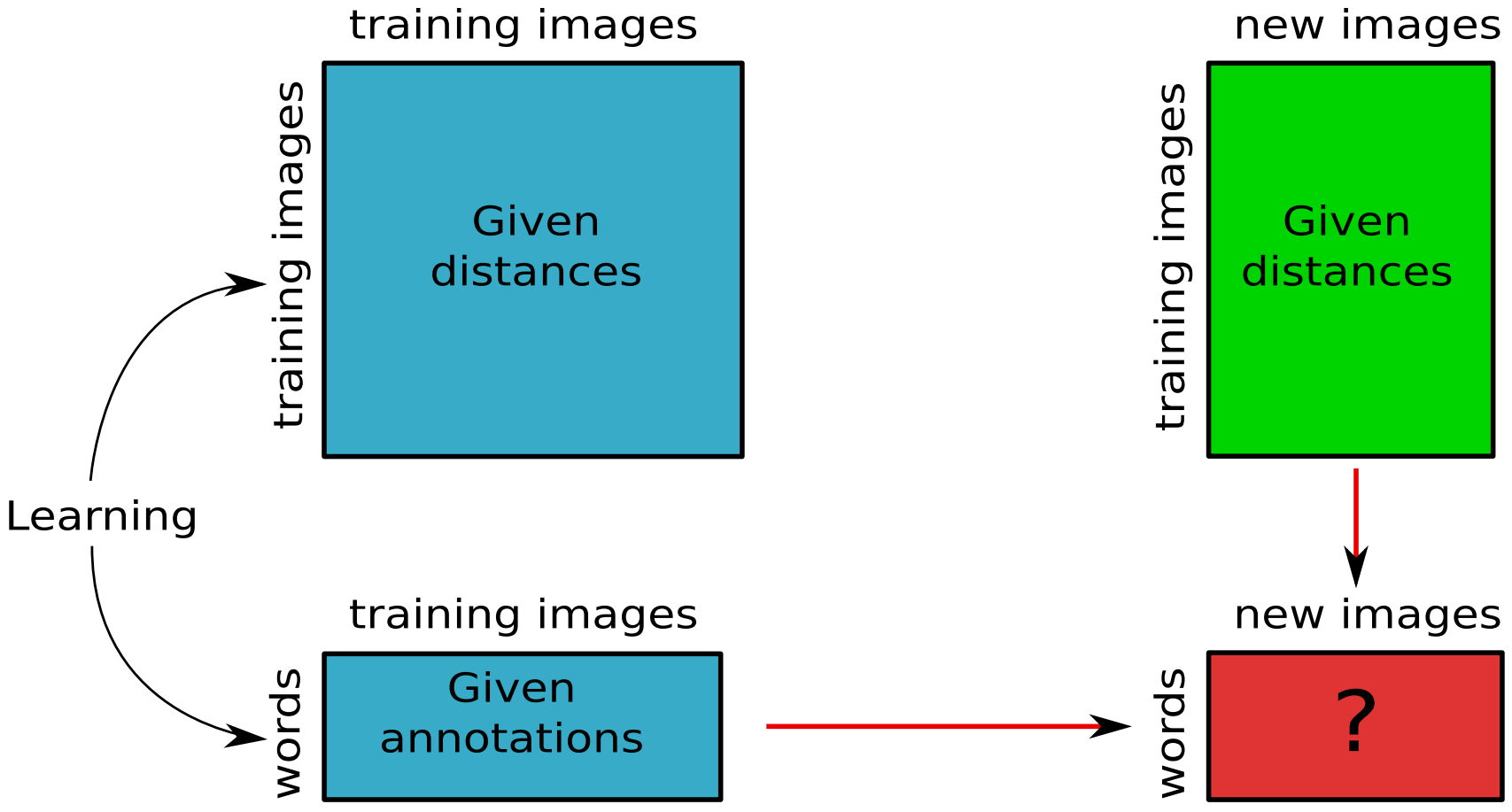
- **What is an appropriate distance to define neighbors?**
  - Which image features to use?
  - What distance over these features?

- **Linear distance combination** defines weights

$$\pi_{ij} = \frac{\exp(-\mathbf{w}^\top \mathbf{d}_{ij})}{\sum_k \exp(-\mathbf{w}^\top \mathbf{d}_{ik})} \tag{8}$$

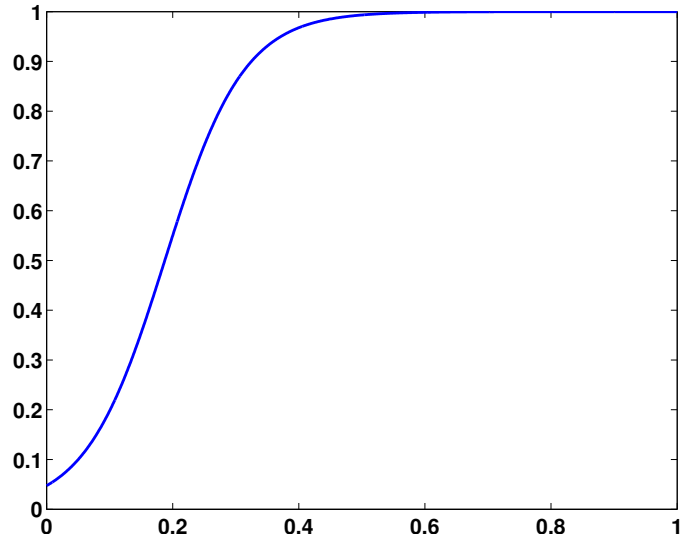
- **Learn distance combination**
  - maximize annotation log-likelihood as before
  - one parameter for each ‘base’ distance

A predictive model for keyword absence/presence



Low recall of rare words

- **Let us annotate images with the 5 most likely keywords**
- **Recall for a keyword is defined as:**
  - # ims. annotated with keyword / # ims. truely having keyword
- **Keywords with low frequency in database have low recall**
  - Neighbors that have the keyword do not account for enough mass
  - Systematically lower presence probabilities
- **Need to boost presence probability at some point**



Sigmoidal modulation of predictions

- **Prediction of weighted nearest neighbor model x_{iw}**

$$x_{iw} = \sum_j \pi_{ij}\, p(y_{iw} = 1|j) \tag{9}$$

- **Word specific logistic discriminant model**
  - Allow to boost probability after a threshold value
  - Adjusts 'dynamic range' per word

$$p(y_{iw} = 1) = \sigma(\alpha_w x_{iw} + \beta_w) \tag{10}$$

$$\sigma(z) = 1/\big(1 + \exp(-z)\big) \tag{11}$$

- **Train model using gradient descent in iterative manner**
  - Optimize (α_w, β_w) for all words, convex
  - Optimize neighbor weights π_{ij} through parameters

Training the model in practice

$$p(y_{iw}) = \sum_j \pi_{ij} p(y_{iw}|j)$$

$$\mathcal{L} = \sum_{i,w} c_{iw} \ln p(y_{iw})$$

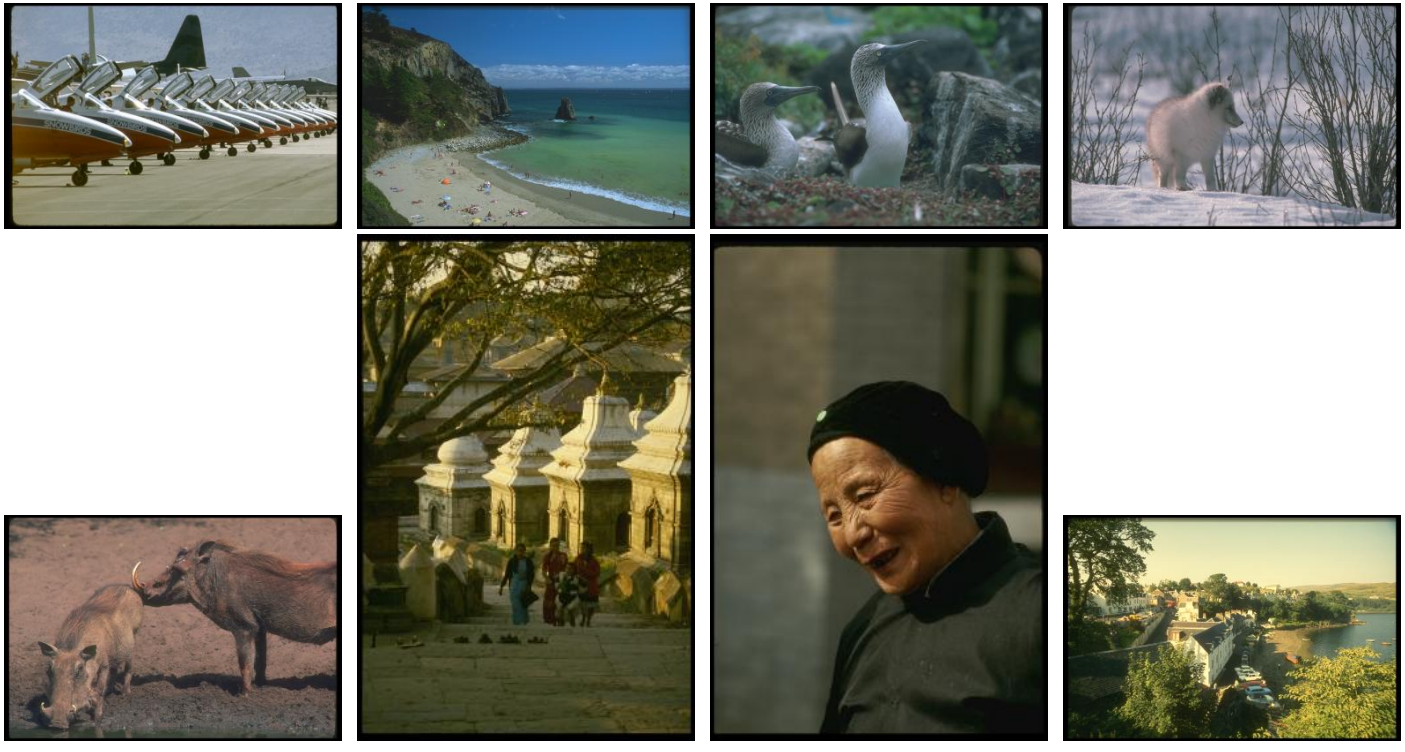
- **Computing $\mathcal{L}$ and gradient quadratic in nr. of images**
- **Use limited set of k 'neighbors' for each image i**
- **We don't know the distance combination in advance**
  - Include as many neighbors from each distance as possible
  - Overlap of neighborhoods allow to use approximately $2k/D$

Presentation Outline

1. Related work
2. Metric learning for nearest neighbors
3. **Data sets & Feature extraction**
4. Results
5. Conclusion & outlook

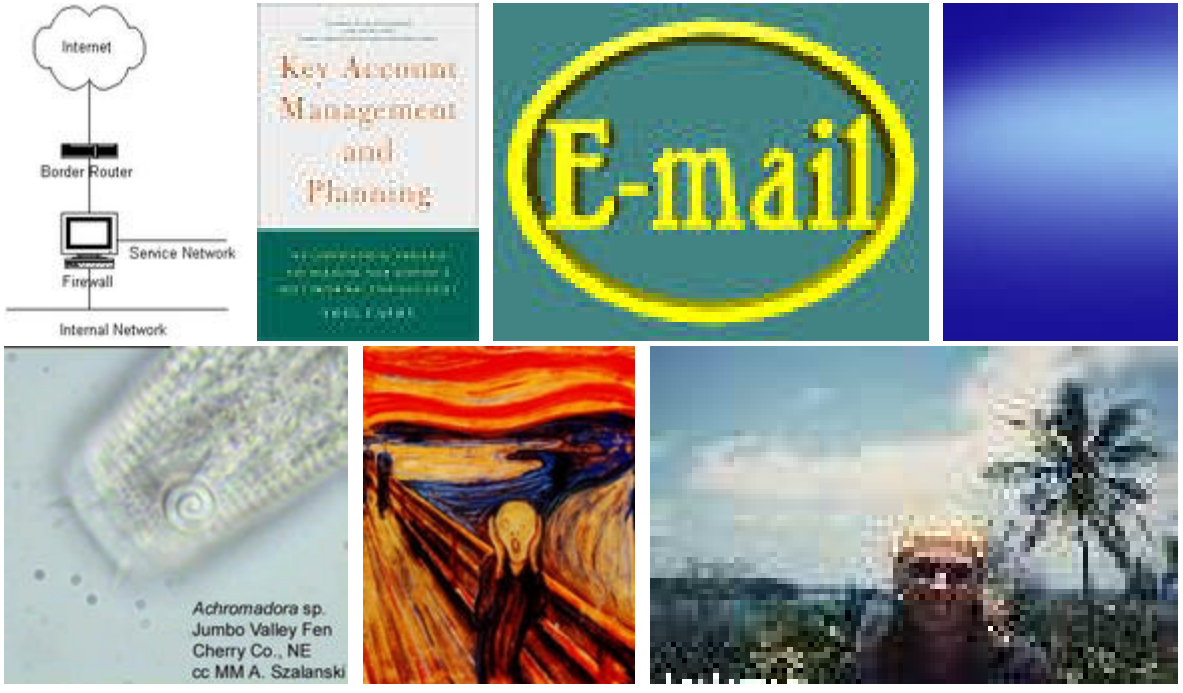
Data set 1: Corel 5k

- **5.000 images**, landscape, animals, cities, . . .
- **3 words on average, max. 5, per image**
- **vocabulary of 260 words**
- **Annotations designed for retrieval**



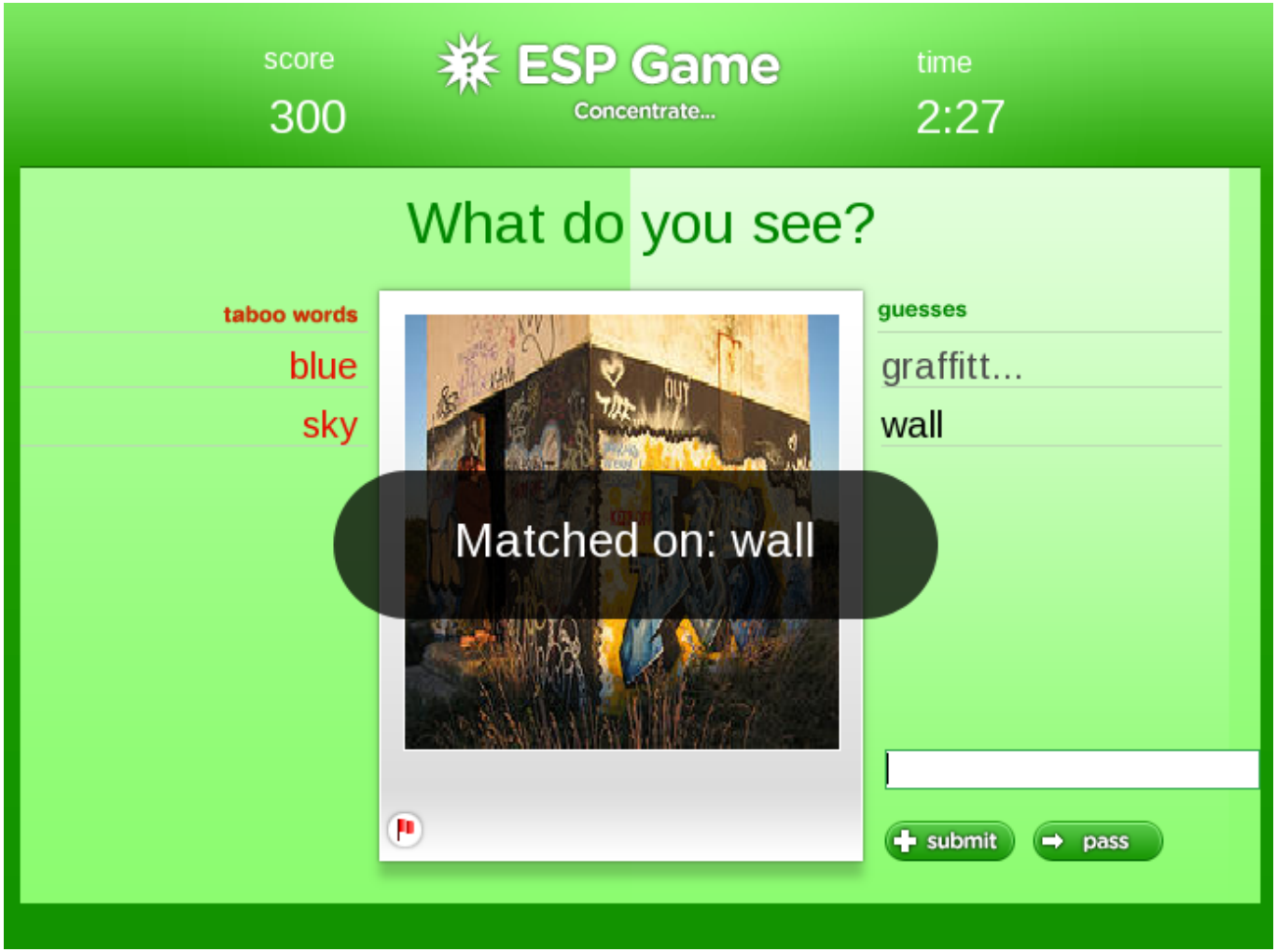
Data set 2: ESP Game

- **20.000 images**, photos, drawings, graphs, . . .
- **5 words on average, max. 15, per image**
- **vocabulary of 268 words**
- **Annotations generated by players of on-line game**



Data set 2: ESP Game

- **Annotations generated by players of on-line game**
  - Both players see same image, but cannot communicate
  - Players gain points by typing same keyword



Data set 3: IAPR TC-12

- **20.000 images**, touristic photos, sports
- **6 words on average, max. 23, per image**
- **vocabulary of 291 words**
- **Annotations obtained from descriptive text**
  - Extract nouns using natural language processing

Feature extraction

- **Collection of 15 representations**

- **Color features**, global histogram
  - Color spaces: HSV, LAB, RGB
  - Each channel quantized in 16 levels
- **Local SIFT features** [Lowe'04]
  - Extraction on dense muti-scale grid, and interest points
  - K-means quantization in 1.000 visual words
- **Local Hue features** [van de Weijer & Schmid '06]
  - Extraction on dense muti-scale grid, and interest points
  - K-means quantization in 100 visual words
- **Global GIST features** [Oliva & Torralba '01]

- **Spatial** 3×1 **partitioning** [Lazebnik et al. '06]
  - Concatenate histograms from regions
  - Done for all features, except GIST.

Presentation Outline

1. Related work
2. Metric learning for nearest neighbors
3. Data sets & Feature extraction
4. **Results**
5. Conclusion & outlook

Evaluation Measures

- **Measures computed per keyword, then averaged**
- **Annotate images with the 5 most likely keywords**
  - **Recall:** # ims. correctly annotated / # ims. in ground truth
  - **Precision:** # ims. correctly annotated / # ims. annotated
  - **N+:** # words with non-zero recall
- **Direct retrieval measures**
  - Rank all images according to a given keyword presence probability
  - Compute precision all positions in the list (from 1 up to N)
  - **Average Precision:** over all positions with correct images

Results Corel - Annotation

	Previsously reported results								Rank Based		Distance Based			
	CRM [10]	InfNet[15]	NPDE [22]	SML [2]	MBRM [5]	TGLM [13]	JEC [14]	JEC-15	WN	σWN	WN	σWN	WN-ML	σWN-ML
P_μ	16	17	18	23	24	25	27	**28**	28	26	30	28	31	**33**
R_μ	19	24	21	29	25	29	32	**33**	32	34	33	35	37	**42**
N+	107	112	114	137	122	131	139	**140**	136	143	136	145	146	**160**

- **Rank-based and distance-based weights comparable**
- **Metric learning improves results considerably**
- **Sigmoid improves recall**

Results Corel - Retrieval

	All	Single	Multi	Easy	Difficult
PAMIR [7]	26	34	26	43	22
WN	32	40	31	49	28
σWN	31	41	30	49	27
WN-ML	**36**	43	**35**	53	**32**
σWN-ML	**36**	**46**	**35**	**55**	**32**

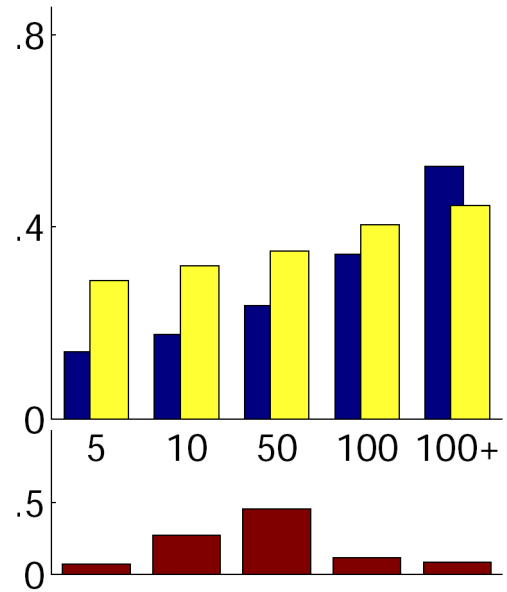
- **Mean Average Precision:** roughly +10% overall
- **Metric learning improves results considerably**
- **Sigmoid small effect:** evaluated per word

Results ESP & IAPR - Annotation

	IAPR			ESP Game		
	P_μ	R_μ	N+	P_μ	R_μ	N+
MBRM [5]	24	23	223	18	19	209
JEC [14]	28	29	250	22	25	224
JEC-15	29	19	211	24	19	222
WN	**50**	20	215	48	19	212
σWN	41	30	259	39	24	232
WN-ML	48	25	227	**49**	20	213
σWN-ML	46	**35**	**266**	39	**27**	**239**

- **Metric learning improves results considerably**
- **Sigmoid:** trades precision for recall

Detailed view of effect sigmoid



- **Mean recall of words**
  - Keywords binned by how many images they occur in
  - WN-ML (blue), and σWN-ML (yellow)
  - The lower bars show nr. of keywords in each bin

Examples - Corel Retrieval

tiger *100.00 (10)*

garden *60.00 (10)*

town *22.22 (9)*

water, pool *90.00 (10)*

beach, sand *25.00 (8)*

Exampels - Corel Annotation

BEP: *100%*
Ground Truth: **sun** *(1.00)*, **sky** *(1.00)*, **tree** *(1.00)*, **clouds** *(0.99)*
Predictions: **sun** *(1.00)*, **sky** *(1.00)*, **tree** *(1.00)*, **clouds** *(0.99)*

BEP: *100%*
Ground Truth: **mosque** *(1.00)*, **temple** *(1.00)*, **stone** *(1.00)*, **pillar** *(1.00)*
Predictions: **mosque** *(1.00)*, **temple** *(1.00)*, **stone** *(1.00)*, **pillar** *(1.00)*

BEP: *50%*
Ground Truth: **grass** *(0.98)*, **tree** *(0.98)*, bush *(0.54)*, truck *(0.05)*
Predictions: flowers *(1.00)*, **grass** *(0.98)*, **tree** *(0.98)*, moose *(0.95)*

BEP: *50%*
Ground Truth: **herd** *(0.99)*, **grass** *(0.98)*, tundra *(0.96)*, caribou *(0.13)*
Predictions: sky *(0.99)*, **herd** *(0.99)*, **grass** *(0.98)*, hills *(0.97)*

BEP: *50%*
Ground Truth: **mountain** *(1.00)*, **tree** *(0.99)*, sky *(0.98)*, clouds *(0.94)*
Predictions: hillside *(1.00)*, **mountain** *(1.00)*, valley *(0.99)*, **tree** *(0.99)*

Break-down of computational effort

- **Computation times for ESP data set** 20.000 images
  - Single recent desktop 4 core machine

- 1h08 : Low-level feature extraction
- 4h44 : Quantization of low level features (best of 10$\times$ k-means)
- 0h22 : Cluster assignments
- 4h15 : Pairwise distances
- 0h15 : Neighborhood computation (2.000)
- 0h05 : Parameter estimation

Conclusions and Outlook

- **We surpassed the current state-of-the-art results**
  - Both on image annotation, and keyword-based retrieval
  - On three different data sets and two evaluation protocols
- **The main contributions of our work**
  - Metric learning within the annotation model
  - Sigmoidal non-linearity to boost recall of rare words
- **Topics of ongoing research**
  - Modeling of keyword absences
  - Learn metric per annotation term
  - Scaling up learning set to millions of images
  - Online learning of the model